Adaptive learning with covariate shift-detection for motor imagery-based brain–computer interface

Adaptive Learning with Covariate Shift-Detection for Motor Imagery based Brain-Computer Interface

Haider Raza*, Hubert Cecotti*, Yuhua Li†, and Girijesh Prasad*
*Intelligent Systems Research Centre, School of Computing and Intelligent Systems, Ulster University, Londonderry, UK.
†School of Computing, Science and Engineering, University of Salford, Manchester, UK.
raza-h@email.ulster.ac.uk, h.cecotti@ulster.ac.uk, y.li@salford.ac.uk , g.prasad@ulster.ac.uk

***Abstract*— A common assumption in traditional supervised learning is that the input data distribution in the training phase follows the same probability distribution as that in testing/operating phase. When transitioning from the training to testing phase, a shift in the probability distribution of input data is known as a covariate shift. Covariate shifts commonly arise in a wide range of real-world systems such as, electroencephalogram (EEG) based brain-computer interfaces (BCIs). In such systems, there is a necessity for continuous monitoring of the process behavior and tracking the state of the covariate shifts to decide about initiating adaptation in a timely manner. This paper presents a covariate shift-detection and -adaptation methodology, and its application to motor imagery based BCIs. A covariate shift-detection test based on an exponential weighted moving average (EWMA) model is used to detect the covariate shift in the features extracted from motor imagery based brain responses. Following the shift detection, the methodology initiates the adaptation by updating the knowledge-base of the classifier during the testing/operating phase. The usefulness of our proposed method is evaluated by using a real-world BCI dataset (i.e. BCI Competition IV dataset 2A and 2B). The results show a statistically significant improvement in the classification accuracy of the BCI system over traditional learning and semi-supervised learning methods.**

***Keywords*—** *Adaptive learning, brain-computer interfaces, covariate shift-detection, transductive learning.*

I. INTRODUCTION

In traditional machine learning techniques, data are assumed to be drawn from stationary distribution. While training a traditional supervised classifier, it is commonly assumed that the input data distribution in the training set and the testing set follows the same probability distribution (Bishop 2006; Duda et al. 2001; Grossberg 1988; Kelly et al. 1998; Mitchell 1997; Vapnik 1999). However, in real-world applications, processes are non-stationary and often characterized by a shifting nature and may shift their distribution over time. In non-stationary environments (NSEs), the data distribution shifts over time; in general this may be due to thermal drift, ageing effects, and noise. The scenario where the training set and testing set follow different distributions but the conditional distribution remains unchanged is known as covariate shift (Sugiyama 2007; Li et al. 2010). In most of the real-world applications, non-stationarity is quite common, especially with the systems interacting with the dynamic and evolving environments, e.g., data coming from electroencephalogram (EEG) based brain-computer interfaces (BCIs), share price prediction in stock market, and wireless sensor networks. Particularly achieving high classification accuracy in a BCI is a challenging task because the signals may be highly variable over time.

A BCI is an alternative communication's means, which allows a user to express his or her will without muscle exertion, provided that the brain signals are properly translated into computer commands (Lotte et al. 2007; Thulasidas et al. 2006; Müller et al. 2004; Wolpaw et al. 2002; Arvaneh, Guan, et al. 2013; Blankertz & Tomioka 2008; Blankertz et al. 2002; Leeb et al. 2007). With an EEG based BCI that operates online in real-time non-stationary/changing environments, it is required to consider input features that are invariant to shifts of the data during long sessions and across sessions, or learning approaches that are able to detect the changes that may repeat overtime, to update the classifier in a timely fashion. The non-stationarities in the EEG may be caused by various reasons such as changing user attention level, electrode placement, and user fatigue (Li et al. 2010; Blankertz & Tomioka 2008; Arvaneh, Cuntai, et al. 2013; Raza et al. 2015). Due to these non-stationarities, it is expected to find notable variations or shifts in the EEG signals during trial-to-trial, and session-to-session transfers (Li et al. 2010; Raza et al. 2013a; Arvaneh, Guan, et al. 2013; Arvaneh, Cuntai, et al. 2013; Blankertz et al. 2002; Blankertz & Tomioka 2008; Leeb et al. 2007; Raza et al. 2015). These variations often appear as covariate shifts in the EEG signals, wherein the input data distributions differ significantly between training/calibration and testing/operating phases, while the conditional distribution remains the same (Raza et al. 2013b; Satti et al. 2010; Sugiyama 2007; Shimodaira 2000; Raza et al. 2014). To date, the low classification accuracy has been one of the main concerns of the developed BCI systems based on a motor imagery (MI) detection, which directly affects the reliability of the BCI (Li et al. 2010; Blankertz & Tomioka 2008; Rezaei et al. 2006). To enhance the performance of BCI systems, several feature extraction, feature selection, and feature classification techniques are proposed in the literature (Shahid & Prasad 2011; Suk & Lee 2013; Kuncheva & Faithfull 2014; Buttfield et al. 2006; Vidaurre et al. 2006; Coyle et al. 2009; Ramoser et al. 2000; Arvaneh, Cuntai, et al. 2013; Arvaneh, Guan, et al. 2013). A large variety of features have been used in MI based BCI such as band powers, power spectral density, time frequency features, and common special patterns (CSP) based features. However, due to brain's non-stationary characteristics, the spatial distribution of the brain responses may change over time, resulting in shifts in feature distributions.

The main drawback of the solutions proposed in the related literature is the requirement of labeled data before starting the adaptation in the evaluation/operating phase (Li et al. 2010; Sugiyama 2012). Additionally, most of the shift-detection methods present in the literature are based on the batch processing for a dataset shift detection (Gama et al. 2013; Alippi et al. 2013; Elwell & Polikar 2011), so there is a time delay in shift-detection. Hence, for real-time systems, the batch processing methods are not beneficial where initiating adaptation in the nick-of-time is of supreme interest. In this paper, we present a novel design methodology for an adaptive classification, which monitors the covariate shift in the input streaming data (i.e., EEG features) through an exponential weighted moving average (EWMA) model based covariate shift-detection (CSD) test (Raza et al. 2013a; Raza et al. 2013b). The CSD test operates in two stages, the first stage deals with covariate shift-detection, and the second stage corresponds to the covariate shift-validation. This two stage structure helps in reducing the false-detection rate, which may reduce an unnecessary retraining of the classifier. An adaptation is only initiated once the covariate shift is confirmed using

validation, in adaptation, the classifier is retrained based on the updated knowledge base ($KB_{Upadated}$) discussed later in Section IV. The proposed method uses different adaptation mechanisms to update the knowledge base (KB_0 i.e., training data) of the classifier on the new knowledge. In the first method, a transductive learning approach is used to add the relevant information in to the KB after each CSD. Moreover, the transductive learning is only used to increase the size of KB, but the overall classification is performed using an inductive classifier. In second method, the KB is updated incrementally using the correctly predicted labels after each CSD. The experiments on the real-world data are used to show that the covariate shift can be adapted using the proposed method. Using the data from the BCI competition-IV 2A and -2B, we have demonstrated that the proposed method can outperform a traditional learning approach and other competing methods.

The novel contributions for the paper can thus be summarized as follows:

- *A covariate shift-adaptation model is introduced to address the effects of non-stationarity in the EEG signals.*
- *An EWMA based CSD test is applied to detect the non-stationary changes in the PCA based features of the motor imagery based brain responses.*
- *The proposed model updates its classification decision boundary online without making any a-priori assumption about the distribution for the upcoming test data.*

This paper proceeds as follows: Section II is a proposed methodology for the covariate shift-detection, -validation, and -adaptation; Section III presents an application to BCI. Section IV shows the results. Finally, Section V gives the discussion.

II. METHODS

A. *Problem Formulation*

Let us consider a learning framework in which training dataset is denoted by $X_{Tr} = \{(x_i, y_i)\}_{i=1}^{N}$, where N is the number of observations, and a target label y_i is associated with each input x_i. Depending upon the number of inputs and outputs, x_i and y_i may be a scalar or vector variables. In the following work, the training dataset is represented as initial knowledge-base (KB_0). Let us consider a two-class classification problem i.e., $y \in \{\omega_1, \omega_2\}$., where $y_i = C_1$, if x_i belongs to ω_1, and $y_i = C_2$, if x_i belongs to ω_2. For example in support vector machine (SVM), we have $C_1 = -1, \text{and } C_2 = +1$. The probability distribution of the inputs at time i can thus be defined as, $P(x_i) = P(\omega_1)P(x_i|\omega_1) + P(\omega_2)P(x_i|\omega_2)$ where $P(\omega_1), P(\omega_2)$ are the prior probabilities of getting a sample of the classes ω_1 and ω_2, respectively, while $P(x_i|\omega_1), P(x_i|\omega_2)$ are the conditional probability distribution for the time period i. The goal is to predict the labels of upcoming samples ($\hat{y}_i$) resulting into $X_{Ts} = \{(\hat{y}_i|x_i)\}_{i=1}^{M}$, where M is the number of observations in the testing phase.

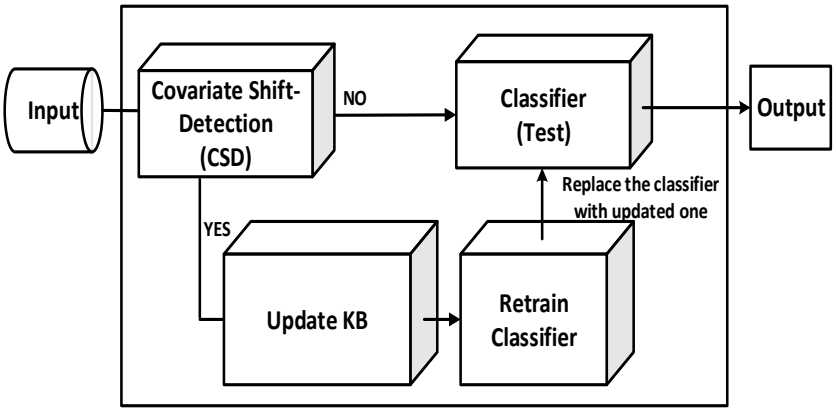


Fig 1. Architecture of the adaptive learning design methodology.

B. Algorithm Overview

The proposed algorithm with the CSD belongs to the category of incremental learning (Elwell & Polikar 2011), where the learning model is updated at each CSD. The covariate shift monitoring is performed using the CSD-EWMA test (Raza et al. 2015; Raza et al. 2013a; Raza et al. 2013b). An advantage of using the CSD test is the enhanced accuracy in terms of low false-positives and low false-negatives. The proposed algorithm is a single classifier based non-stationary learning (NSL) algorithm that uses the CSD-EWMA test for initiating adaptive corrective action. The algorithm is provided with a time-series training dataset KB_0, where $KB_0 = X_{Tr}$, and a classifier $\mathcal{F}$ is trained. In the evaluation phase, the CSD-EWMA test is used to monitor and detect the covariate shift and the classifier $\mathcal{F}$ is then used to classify the upcoming input data X_{Ts}.

The key elements of the proposed solution are:

- *CSD*: CSD test monitors the stationarity of x_i, disregarding their supervised labels.
- $\mathcal{F}$: The pattern classifier $\mathcal{F}$ is used to classify the input samples.
- $KB_{Updated}$: Updated knowledge base ($KB_{Updated}$) i.e., updated on each CSD.

Algorithm 1: Learning with CSD

1. Configure the classifier $\mathcal{F}$ based on the initial knowledge base $KB_0 = X_{Tr}$;
2. Configure the parameters λ and L for the CSD test using the KB_0 ;
3. $KB_{Updated} = KB_0$
4. **FOR** $i = 1$ to length(X_{Ts})
5. Receive new data x_i;
6. **IF** (CSD detects and validates a non-stationarity at time i), **THEN**
7. $KB_{Updated} \leftarrow KB_{Updated} \cup KB_{New}$
8. Retrain and adapt the classifier $\mathcal{F}$ on $KB_{Updated}$
9. **END**
10. Classify the input x_i by the classifier $\mathcal{F}$ and get the predicted label $\hat{y}_i$;
11. **END**

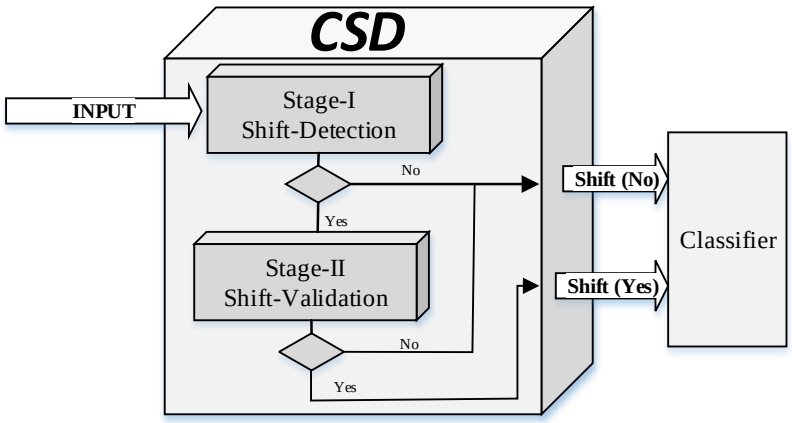


Fig. 2: A two-stage covariate shift-detection (CSD). Stage-I is for shift-detection and stage-II works for validation.

The proposed solution is described in Algorithm 1. After a preliminary configuration phase of the base classifier $\mathcal{F}$ and CSD on the $KB_{0,}$ the CSD is used to assess the process stationarity. As soon as the CSD detects a covariate shift in the upcoming unlabeled data, the classifier learned model becomes obsolete, and has to be replaced with a newly configured/retrained model. At each CSD, the new information (i.e., KB_{New}) becomes available containing the information about the new data distribution. Next, the KB_{New} is merged with existing $KB_{Updated}$, and a new $KB_{Updated}$ is prepared. To prepare the $KB_{Updated}$, two methods are identified based on computational intelligence techniques: first is a transductive learning with CSD (TLCSD), and second is an adaptive learning with CSD (ALCSD). The interactions between the covariate shift-detection, -validation, and -adaptation stages are more clearly illustrated with the help of Fig. 1 and Fig. 2, which are explained in the following subsections.

C. Covariate Shift-Detection (CSD)

The first step requires in a CSD test is to detect the covariate shift in the process, possibly without relying on the prior information about the process data distribution before and after the shift. This is a crucial step for reconfiguring the classifier, and it acts as an alarm. Since this test has to be executed online, its computational complexity may be a critical issue. The first-stage of the test provides an initial estimate of the shift (i.e., where the actual shift has occurred). The first-stage test is performed by an SD-EWMA (Raza et al. 2013a) based test. If the test outcome at the first-stage is positive, then the second stage test gets activated, and a validation is performed in order to reduce the number of false-alarms (Raza et al. 2013b). The second stage test/validation procedure is discussed in next sub-section. The choice of the smoothing constant λ and a control limit multiplier (L) are the important issue in the EWMA-CSD test. The choice of λ and L are discussed in Section IV.

In an EEG-based BCI, the EEG data are generated from multiple electrodes, and after applying a feature extraction technique there are multiple features obtained, and hence BCI input data are multivariate. Monitoring of such input processes independently may be misleading, e.g., if the probability that a variable exceeds three-sigma control limits is 0.0027, then a false-detection rate of 0.27% is expected. However, the joint probability that d such

variables exceed their control limits simultaneously is $(0.0027)^d$, which is considerably smaller than 0.0027. So, the use of d-independent charts may provide highly distorted outcomes. A principal component analysis (PCA) is therefore used to reduce the dimensionality of the data (Rosenstiel et al. 2012; Kuncheva & Faithfull 2014). It provides fewer components, containing most of the variability in the data. We have used a single component to monitor the shift in the process using SD-EWMA test (Raza et al. 2013a) at the first stage .

D. Covariate Shift-Validation

According to the Algorithm 1, the KB of the classifier has to be updated at each CSD. However, false positives (i.e., detection that does not correspond to a true shift in the input distribution) result in an unnecessary retraining. To counter this, we have introduced a covariate shift-validation procedure as part of a two-stage structure test (Raza et al. 2013b). This strategy aims at guaranteeing that the classifier relies on an up-to-date KB, and the classifier is only retrained on the occurrence of a valid shift. The covariate shift-validation procedure exploits two sets of observations generated before and at the CSD time point. The observations from the KB_0 are assumed to be in its stationary state, and are compared with data from the current trial, at the CSD time point. To validate the CSD from the stage-I, a multivariate Hotelling's T-Square statistical hypothesis test is used (Hotelling 1947). If the p-value of the test is below 0.05, then the CSD is confirmed, otherwise it is considered as a false-alarm. On each CSD, the KB_{New} is obtained based on the current shift in the data.

E. Covariate Shift-Adaptation

Once the CSD is validated, the adaptation phase starts (see Fig. 2). To adapt to the shift, re-training of the classifier is required. In order to retrain the classifier, an additional set of input target pairs is necessary to prepare the $KB_{Updated}$. To get the set of input target pairs, we have investigated two ways for the KB management based on computational intelligence techniques. In the first scenario (i.e., TLCSD), we have applied a transductive-inductive learning model to adapt to a covariate shift. However, transduction part is only used to add new trials into the KB_{New}, and an inductive classifier is used to classify the upcoming samples from the evaluation phase. The transduction will only start once the covariate shift is detected and validated. In the second scenario (i.e., ALCSD), it is assumed that during the evaluation phase, a true label is available after each trial. Once the covariate shift is detected then only correctly predicted labels are added into KB_{New}, and the classifier is re-trained, and the updated classifier is used for further classification. This approach is quite similar to co-training (Zhu 2008) used in a semi-supervised learning (SSL), where the predicted labels are used to train the other classifier.

Both the methods mentioned above used to adapt to the covariate shift are presented below.

1) *Transductive Learning with CSD (TLCSD)*

A TLCSD model is based on a probabilistic K-nearest neighbor (KNN) method. Initially according to Algorithm 1, at step 1, an inductive classifier $\mathcal{F}$ is trained on the KB_0, and at step 2, the parameters λ and L are set for the CSD test. Once the classifier $\mathcal{F}$ is trained, then an evaluation phase starts. At step 3, the KB_0 is assigned to $KB_{Updated}$, and

the parameters λ, L, CR_{Thres}, and K are set, wherein CR_{Thres} is a confidence ratio threshold that is used to decide the usefulness of the trial, and K is the number of neighbors for the transductive learning. In the evaluation phase, the classifier takes the features as the input obtained from the testing data. The classifier initiates adaptation through transduction after every CSD. Each time the classifier initiates adaptation at step 7, it is considered as one epoch and it takes Δm data points to predict the labels through a transductive function $\mathcal{T}$, where Δm is the number of points between two shift-detection points or from the start of evaluation phase to the first detection point. Once the adaptation is initiated at each epoch, the Euclidean distance ($d_{p,q}$) from the unlabeled data point x_p to the labeled data point x_q is computed as given below:

$$d_{(p,q)} = \|x_p - x_q\| \tag{1}$$

This provides a vector **D**= $[d_{(p,q_1)}, \dots, d_{(p,q_N)}]$ of Euclidean distances from unlabeled data point to the N number of labeled data points. Then, the K nearest neighbors are selected. For each of the K nearest points, an RBF kernel is used to compute the weight, as given in equation (2).

$$K(p,q) = exp\left(-\frac{\|x_p - x_q\|^2}{2\sigma^2}\right) \tag{2}$$

From equation (2), we have $0 \leq K(p,q) \leq 1$. A weight with a high value implies the data-point's closeness to the unlabeled current feature. Thus the weight for each neighbor is given by,

$$R(i) = K(p,q_i) \tag{3}$$

Using $R(i)$ and the existing KB, for each of the classes a confidence ratios CR_{ω_i} is obtained by,

$$CR_{\omega_1} = P(\omega_1|x) = \frac{\sum_{i=1}^{K} R(i) * (y(i) == \omega_1)}{\sum_{i=1}^{K} R(i)} \tag{4.a}$$

$$CR_{\omega_2} = P(\omega_2|x) = \frac{\sum_{i=1}^{K} R(i) * (y(i) == \omega_2)}{\sum_{i=1}^{K} R(i)} \tag{4.b}$$

The confidence ratio CR_{ω_i} attained from equation (4.a & 4.b) may be viewed as a posterior probability of the class membership of the current unlabeled data point, as $CR_{\omega_1} + CR_{\omega_2} = 1$. This CR_{ω_i} acts as a belief or confidence, which determines if a data sample belongs to a particular class. In this step, for each observation from Δm, CR_{ω_i} is obtained and it is used to decide if the trial's features and the estimated output labels should be added to the existing knowledge-base i.e. if $\max(CR_{\omega_1}, CR_{\omega_2}) > \mathrm{CR}_{\mathrm{Thres}}$, then the example is stored into KB_{New}, otherwise it is discarded. At step 7, this KB_{New} is then merged into the existing $KB_{updated}$. Based on the updated KB, the inductive

classifier function is updated, and a new classifier $\mathcal{F}$ is obtained at step 8. Every time KB_{New} is available, the classifier $\mathcal{F}$ is updated, and this process is repeated until all the *M* points in the testing phase are classified.

2) ***Adaptive Learning with CSD (ALCSD)***

In ALCSD, initially at step 1 of Algorithm 1, an inductive classifier $\mathcal{F}$ is trained on the $KB_{0.}$ The KB_0 consists of *N* labeled trials. Using KB_0 at step 2, the parameter λ is obtained for the CSD test, and the control limit (L) for the CSD is set to L=2. Then, an evaluation phase starts at step 4, and unlabeled features from X_{Ts} are processed sequentially for classification. At step 6, the CSD test is used to monitor the covariate shift. Once the covariate-shift is detected, it acts as an alarm to update the classifier. To update the classifier, new knowledge from the data is required. In order to obtain a KB_{New}, it is assumed that in each trial, the true label is available, and among all predicted labels only correctly predicted labels through an inductive classifier are added into $KB_{New.}$ The $KB_{updated}$ and KB_{New} are merged at step 7 to form a $KB_{Updated}$. The $KB_{Updated}$ is used to retrain the classifier at step 8, and further at step 10, this updated classifier is used to classify the upcoming data. On each CSD, the KB thus gets updated and a classifier is built and adapted incrementally.

III. Application to Brain-Computer Interface

1) ***Data Description***

a) BCI Competition IV dataset 2A

The BCI Competition IV dataset 2A (Blankertz n.d.; Tangermann et al. 2012)is comprised of the EEG data collected from nine subjects, namely [A01-A09], that were recorded during two sessions on separate days for each subject. The data consists of 25 channels, which include 22 EEG channels, and 3 monopolar EOG channels. Among the 22 EEG channels, 10 channels are selected for this study, which are responsible for capturing most of the motor imagery activities. The selected channels are presented in Fig. 3(a). The data was collected on four different motor imagery tasks: left hand (class 1), right hand (class 2), both feet (class 3), and tongue (class 4). Each session consists of six runs separated by short breaks, each run comprised of 48 trials (12 for each class). The total numbers of 288 trials are in each session. Only the class 1 and the class 2 for left hand and right hand were considered in this study (i.e., 144 trials). For more details about the dataset kindly refer to (Brunner et al. 2008). The motor imagery data from the session-I was used to train the classifiers, and the motor imagery data from the session-II was used as the test data-set.

b) BCI Competition IV dataset 2B

BCI competition 2008-Graz dataset B (Blankertz n.d.) is a dataset consisting of EEG data from 9 subjects, namely [B01-B09]. Three channel bipolar recordings (C3, Cz and C4) were acquired with a sampling frequency of 250 Hz, the montage is shown in Fig. 3(b). All signals were recorded monopolarly with the left mastoid serving as reference, and the right mastoid as ground. For each subject, five sessions are provided. The motor imagery data from session-I & -II were used to train the classifiers, the data from session-III was used to obtain the hyperparameters (i.e., K and

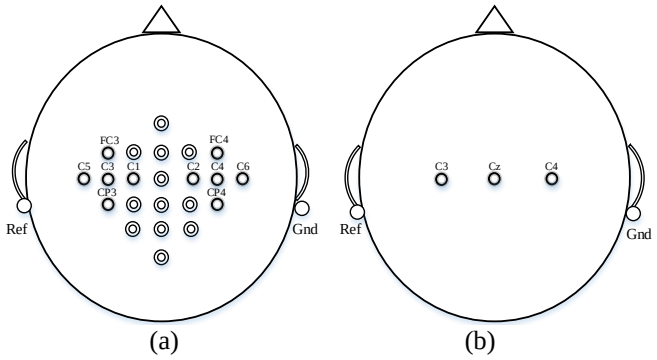


Fig. 3. Electrode montage corresponding to the international 10-20 system: (a) Dataset 2A, among all 22 EEG channels, total 10 channels are selected as shown in black filled hollow circles. (b) Dataset 2B, all channels are selected.

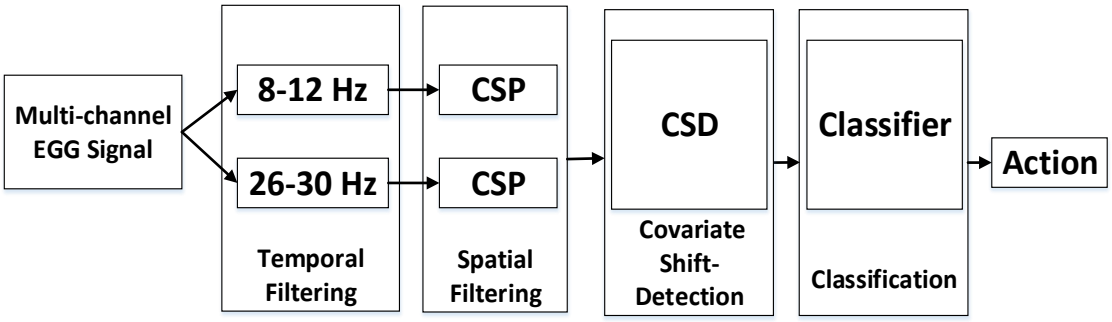


Fig.4: Block diagram for the MI based BCI. It consist of following five stages: Initially the multichannel EEG signals are acquired, next the band-pass filtering is performed, and then the CSP features are obtained, and the covariate shift is monitored, and then features are classified using a pattern classifier. Finally, the action is performed.

CR_{Thres}), and the motor imagery data from session-IV & -V were used to evaluate the performance of the test. Session-IV & -V consist of 160 trials each. Each trial is a complete paradigm of 8 seconds, for more details refer to (Leeb et al. 2008).

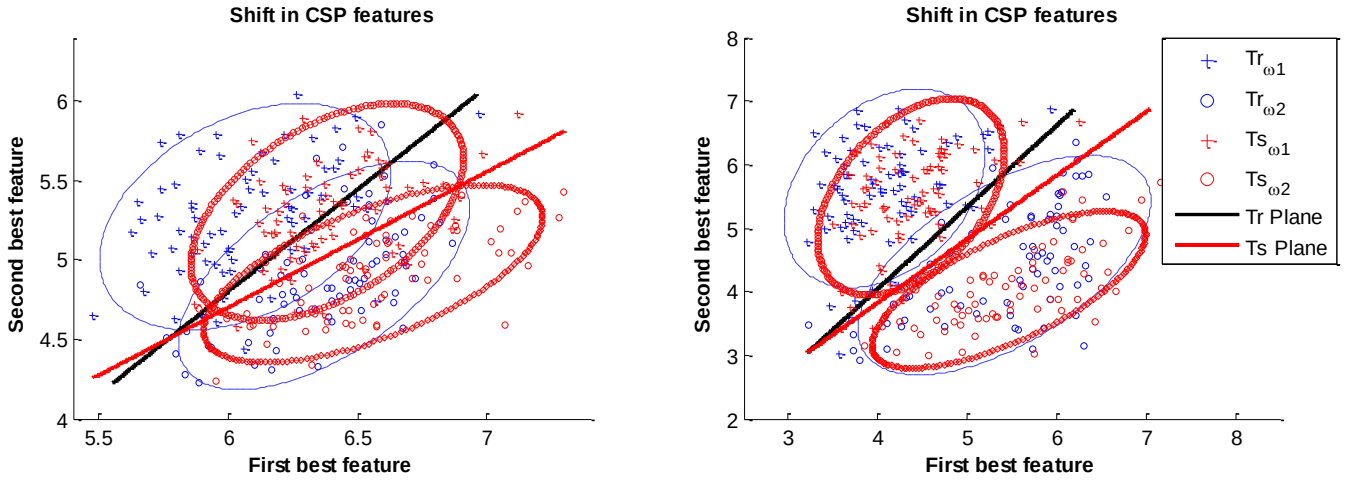
2) ***Data Processing and Feature Extraction***

a) Temporal Filtering

The second stage of the MI based BCI block diagram (see Fig. 4) employs two filters that decomposes the EEG signals into two different frequency bands. The band-pass filters are used, namely [8-12] Hz (μ band), [14-30] Hz (β band). These frequency ranges are used because they cover a stable frequency response related to MI associated phenomena of event-related synchronization and de-synchronization (ERS/ERD). In the next sections, we consider a time segment of 3 s after the cue onsets for both data sets.

b) Spatial Filtering

The third stage employs a spatial filter that maximizes the variance of spatially filtered signals under one condition, while minimizing it for the other condition. Raw EEG scalp potentials are known to have poor spatial resolution due to volume conduction. If the signal of interest is weak while other sources produce strong signals in the same frequency range, then it is difficult to classify two classes of EEG measurements (Blankertz & Tomioka 2008). The neurophysiological background of motor-imagery based BCIs is that motor activity, both actual and imagined, causes an attenuation or increase of localized neural rhythmic activity called Event-Related



(a) [8-12] Hz (b) [14-30] Hz

Fig. 5: Covariate shift in the EEG dataset 2A-subject A03, between training and testing input distribution for different frequency bands. (a) Mu band [8-12] Hz, and (b) Beta band [14-30]Hz. The red circle denote the features of the left hand motor imagery and blue crosses denote the features of the right hand motor imagery. The black and red lines represent the decision boundaries obtained by the training data and test data respectively.

Desynchronization (ERD) or Event-Related Synchronization (ERS). The Common-Spatial-Pattern (CSP) algorithm is highly successful in calculating spatial filers for detecting (ERD/ERS) (Ang et al. 2012; Ang et al. 2008). The objective of CSP algorithm is to compute features whose variances are optimal for discriminating two classes of EEG.

A pair of band-pass and spatial filters in the first and second stages perform spatial filtering of EEG signals that have been band-pass filtered in a specific frequency range. Thus, each pair of band-pass and spatial filter computes the CSP features that are specific to the band-pass frequency range. CSP is a technique to analyze multichannel data based on the recording from two classes (Blankertz & Tomioka 2008). It is a data driven supervised decomposition of signals parameterized by a matrix $\boldsymbol{W} \in \mathbb{R}^{C\times C}$ (C: number of selected channels) that projects the single trial EEG signal $\boldsymbol{E} \in \mathbb{R}^{C}$ in the original sensor space to $\mathbf{Z} \in \mathbb{R}^{C}$, which lives in the surrogate sensor space, as follows:

$$\mathbf{Z} = \boldsymbol{W}\boldsymbol{E} \qquad (5)$$

where $\boldsymbol{E} \in \mathbb{R}^{C\times T}$ is a EEG measurement data of single trial, C is the number of channels; T is the number of samples per channel. W is the CSP projection matrix. The rows of the $\mathbf{W}$ are the spatial filters and the columns of W are the common spatial patterns.

The spatial filtered signal $\mathbf{Z}$ given in eq. (5) maximizes the difference in the variance of the two classes of EEG measurements. A CSP analysis is applied in order to obtain an effective discrimination of mental states that are characterized by ERD/ERS effects. However, the variances of only a small number (m) of the spatial filtered signal are generally used as feature for classification. The m first and last rows of $\mathbf{Z}$ i.e. $\boldsymbol{Z_t}$, t$\in \{1 \dots 2m\}$ form the feature vector $\boldsymbol{x_t}$ given by,

$$x_t = log\left(\frac{var(Z_t)}{\sum_{i=1}^{2m} var(Z_i)}\right) \quad (6)$$

Here, m=1. The CSP features from both frequency bands are combined to form the input features for the training a single classifier. Fig. 5 shows the covariate shift in the CSP features for training and test datasets for subject A03 over two different frequency bands mu (μ) [8-12] Hz and beta (β) [14-30] Hz. The blue crosses and red circles denote the features of the left hand and right hand motor imagery, respectively. The black line and red line represent the separation planes between the features of two classes obtained from two frequency bands as training and testing features, respectively. The separation planes are plotted for illustration purpose only.

3) *Covariate Shift-Detection (CSD)*

The fourth stage uses the CSD test on the CSP features. In both datasets, the data are generated from multiple channels and for each channel two features are produced from each frequency band. To use the CSD-EMWA, the PCA is used to reduce the dimensionality of the features and a single component is used to detect the covariate shift. To execute the CSD test, the smoothing constant λ is selected for each subject based on minimizing the sum of squares of 1-step-ahead prediction error method, and the control limit multiplier is set to L=2. The choice of L has a major impact on the performance of the CSD test, a small value of L makes it more sensitive in detecting the minor shifts in the data. The CSD test in the operational stage detects the shift and validates it through its two stage structure. If the CSD test is positive then a classifier is retrained on the $KB_{Updated}$, and a new classification decision boundary is obtained.

4) *Classification and Evaluation Metrics*

In order to evaluate the performance of the system, we have considered the classification accuracy as the measure of index. The experiments are performed using a support vector machine (SVM) pattern classifier $\mathcal{F}$. The accuracy is given in percentage (%). In CSD tests, the percentage (%) of covariate shift-detected and shift-validated are computed as given below:

$$\%\ of\ shift\ detected/validated = \left(\frac{(\#shift\ detected/validated)}{Total\ number\ of\ trials}\right) \times 100 \quad (7)$$

The parameters K and CR_{Thres} are required to be carefully selected for the TLCSD. In the dataset 2A, the session-I is divided into two parts, first 80% for training the pattern classifier and second 20 % is used to obtain the optimized parameters. The evaluation is then performed on the data from session-II. In the dataset 2B, the session-I & session-II are used for training the pattern classifier, the session-III is used to obtain the optimized parameters, and the session-IV & the session-V are used to evaluate the performance of the classifier. For each dataset, a 10-fold cross-validation (10-CV) accuracy on the training data is computed. Moreover, the two variants for the proposed methods are evaluated and compared with a baseline method and a label propagation based semi-supervised learning (SSL) algorithm. The baseline method uses an inductive learning classifier with CSP features (Ramoser et al. 2000), but it does not adapt/re-train its pattern classifier. To compare with a similar method, the SSL with label propagation

TABLE I
RESULTS FOR SHIFT-DETECTION & VALIDATION DATASET 2A.

Subject	Lambda	Shift-Detected (%)	Shift-Validated (%)
A01	0.10	7.64	2.78
A02	0.80	7.64	6.25
A03	1	4.86	1.39
A04	1	7.64	4.17
A05	0.30	10.42	4.17
A06	0.10	9.72	3.47
A07	0.10	8.33	6.25
A08	0.20	7.64	3.47
A09	0.50	6.94	2.78

TABLE II
CLASSIFICATION ACCURACY (%) RESULTS FROM BCI COMPETITION IV-DATASET 2B.

	10- CV	Baseline	SSL	$TLCSD_1$	$TLCSD_2$	ALCSD	UB
	Tr	**Eval**	**Eval**	**Eval**	**Eval**	**Eval**	**Eval**
A01	85.71	89.58	79.17	90.28	90.28	90.28	90.28
A02	75.71	53.47	54.17	57.64	57.64	54.17	58.33
A03	92.86	92.36	93.06	93.06	95.14	93.75	97.22
A04	77.86	64.58	68.06	65.28	65.97	64.58	67.36
A05	61.43	59.03	45.14	59.72	61.11	57.64	59.03
A06	71.43	65.28	56.94	65.28	65.28	65.28	65.97
A07	84.29	59.72	54.17	59.72	61.11	62.50	70.83
A08	93.57	91.67	90.97	90.28	91.67	90.97	90.97
A09	80.00	85.42	87.50	85.42	86.11	85.42	90.28
Mean	**80.32**	**73.46**	**69.91**	**74.07**	**74.92**	**73.84**	**76.70**
Std	**10.25**	**15.94**	**18.22**	**15.21**	**15.43**	**15.93**	**15.33**
***p-Value**			**0.3047**	**0.2813**	**0.0156**	**0.5313**	**0.0156**

*A two-sided Wilcoxon signed rank test is used to assess the statistical significance of the improvement at a confidence level of 0.05, the p-value denotes the Wilcoxon signed rank test

method (Zhu & Ghahramani 2002) has been considered. A two-sided Wilcoxon signed rank test is used to assess the statistical significance of the pairwise comparison at a confidence level of 0.05.

IV. RESULTS

1) *Results for Dataset 2A*

The results for the choice of the smoothing constant λ and the CSD are presented in Table I. The value of λ is obtained by minimizing the sum of squares of 1-step-ahead prediction errors. The covariate shift was detected maximum number of times for the subject A05 (i.e. 10.42% CSD) and minimum number of times for the subject A03 (i.e. 4.86% CSD). After the covariate shift-validation stage, for the subject A05, the CSD has-been decreased from 10.42% to 4.17%, and for the subject A03, the CSD has-been decreased from 4.86% to 1.39%. The validation stage thus helps to decrease the rate of false-positive at stage-II; consequently the effort of unnecessary retraining the classifier is also reduced.

In Table-II, the 10-CV gives the average training accuracy of 80.32±10.25%, while the subject A08 has a maximum accuracy of 93.57%. For the baseline results, an inductive classifier is used for the classification on the test data without any adaptation on the CSP features. The baseline method gives an average accuracy of 73.46±15.94% and

TABLE III
RESULTS FOR SHIFT-DETECTION & VALIDATION DATASET 2B.

Subject	Lambda	Shift-Detected (%)	Shift-Validated (%)	Shift-Detected (%)	Shift-Validated (%)
		Session IV		Session V	
B01	0.10	10.00	4.38	6.88	3.13
B02	0.80	6.88	1.25	9.38	5.63
B03	1	6.88	2.50	8.13	6.25
B04	1	1.88	0.63	3.75	1.25
B05	0.30	7.50	4.38	6.88	3.75
B06	0.10	8.13	5.00	10.00	6.88
B07	0.10	6.25	5.63	7.50	2.50
B08	0.20	6.25	4.38	10.00	5.00
B09	0.50	8.13	4.38	8.13	3.75

TABLE IV
CLASSIFICATION ACCURACY (%) RESULTS FROM BCI COMPETITION IV-DATASET 2B.

	10- CV	Baseline	SSL	$TLCSD_1$	$TLCSD_2$	ALCSD	UB
	Tr	Eval	Eval	Eval	Eval	Eval	Eval
B01	70.42	69.69	66.56	69.06	70.31	71.88	75.00
B02	61.25	49.58	51.56	50.00	50.63	50.00	51.56
B03	56.67	51.56	49.38	48.44	52.81	52.81	52.19
B04	88.85	93.13	85.63	93.44	93.75	93.44	96.56
B05	76.15	52.81	51.25	62.81	63.75	54.37	77.19
B06	70.71	72.81	67.50	72.19	74.06	73.13	74.06
B07	84.29	58.13	56.25	59.38	61.88	62.50	70.00
B08	61.79	65.63	64.38	65.63	83.13	77.81	88.44
B09	66.25	73.75	72.19	74.38	77.19	75.00	75.00
Mean	**70.71**	**65.23**	**62.74**	**66.15**	**69.72**	**67.88**	**73.33**
Std	**10.78**	**13.98**	**11.89**	**13.64**	**14.05**	**14.16**	**14.67**
p-Value			**0.0391**	**0.6719**	**0.0039**	**0.0039**	**0.0039**

*A two-sided Wilcoxon signed rank test is used to assess the statistical significance of the improvement at a confidence level of 0.05, the p-value denotes the Wilcoxon signed rank test

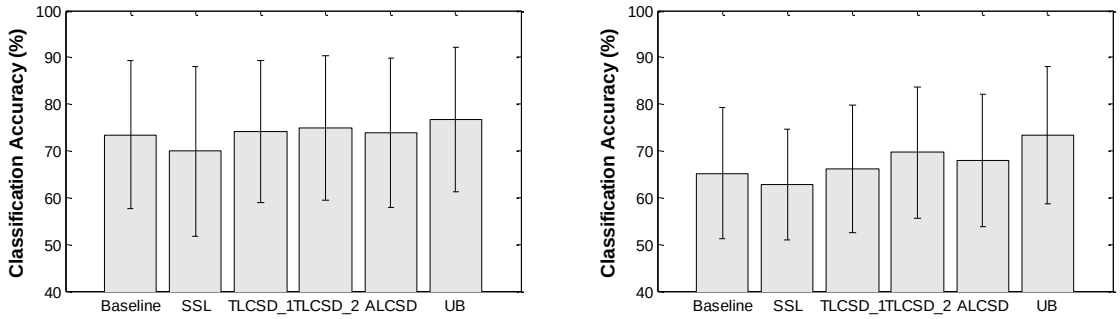
the subject A03 has the highest accuracy 92.36%. The SSL based label propagation method gives an average accuracy of 69.91±18.22%, which is inferior to the baseline method. In $TLCSD_1$, the parameters K and CR_{Thres} have been set to K=18 and CR_{Thres}=0.70, and the classification accuracy has improved slightly from 73.46±15.94% to 74.07±15.21%.

For $TLCSD_2$, the subject specific parameters are selected based on a grid search hyperparamter optimization technique and the all subjects have shown an improvement except the subject A08. The average accuracy for the $TLCSD_2$ has improved from 73.46±15.94% to 74.92±15.43% (p-value= 0.0126). In the ALCSD method, the results have shown a slight improvement in the performance against the baseline method with the mean accuracy of 73.84±15.93%; only subjects A01, A02, A03, and A07 have shown improvement. In the last column of the Table-II, the maximum classification accuracy is obtained by training the pattern classifier on both the train and the test data, and evaluated on test data. Thus the mean classification accuracy of 76.70±15.33% represents the performance that can be achieved if all the data is available for training. Fig. 6 depicts the average classification accuracy across subjects.

2) *Results for Dataset 2B*

The results for the choice of λ and the CSD are presented in Table III. In this dataset, session IV and V are used for evaluation phase, hence for each session the CSD test is performed independently. In session IV, the subject B01 has the maximum number of CSD (10%), and subject B04 has minimum number of CSD (1.88%). After the covariate shift-validation stage, for the subject A01, the number of CSD has-been decreased from 10% to 4.38%, and for the subject A04, the number of CSD has-been decreased from 1.88% to 0.63%. Moreover, in session V, the subjects B06 and B08 have the maximum number of CSD (10%), and subject B04 has minimum number of CSD (3.75%). After the covariate shift-validation stage, for the subject B06, the number of CSD has-been decreased from 10% to 6.88%, and for the subject B08, the number of CSD has-been decreased from 10% to 5%.

In Table-IV, the 10-CV gives the average training accuracy of 70.71±10.78%, where the subject B04 has a maximum accuracy of 88.85%. The baseline method gives 65.23±13.98% of average accuracy and subject B04 has the maximum accuracy 93.13%. The SSL based label propagation method gives 62.74±11.89% average accuracy, which is below the accuracy of the baseline method. In $TLCSD_1$, the parameters K and CR_{Thres} have been fixed to K=18 and CR_{Thres}=0.70, and the classification accuracy has slightly improved from 65.23±13.98% to 66.15±13.64%. Next, for $TLCSD_2$, the subject specific parameters are selected based on the grid search hyperparamter optimization technique and all the subjects have shown an improvements. The average accuracy for the $TLCSD_2$ has improved from 65.23±13.98% to 69.72±14.05% that is statistically significant better (p-value=0.00039). Next, in ALCSD method, the results have shown a considerable improvement in the performance against the baseline method with the mean accuracy of 67.88±14.16%, which is statistically significant better with (p-value of 0.0039). Moreover, for ALCSD, all the subjects have shown an improvement. In the last column of the Table-IV, the maximum classification accuracy is obtained by training the classifier on both the training and the test data, and evaluated on test data, leading to the average classification accuracy of 73.33±14.67%. Fig. 6. (b) Represents the average classification accuracy across subjects.



(a) (b)

Fig. 6: Comparison of the mean accuracies for the proposed methods against the baseline, SSL, and UB on (a) the dataset 2A and (b) dataset 2B.

V. DISCUSSION

The proposed TLCSD and ALCSD methods for the EEG-based BCI are based on a covariate shift-detection and an adaptation framework. An EWMA-CSD test is used to detect the covariate shift. Once the shift is detected an appropriate adaptive action is initiated to address the effect of the covariate shift. In TLCSD, the new information/knowledge obtained through transduction is used to update the KB (i.e., training data) of the inductive classifier. However, the global classification function is still inductive because the transductive knowledge is only used to add more information into KB.

An important issue in the CSD is the choice of the control limit multiplier L. Considering small limit L=2 means focusing on minor shifts, such as muscular artifacts arising during trial-to-trial transfer. However, the long term non-stationarities may be accounted for by considering a large value of L=3, such as during session-to-session transfer or run-to-run transfer. We have selected a small value of control limit multiplier L=2, as our aim is to detect the covariate shift that arises during trial-to-trial transfers. The proposed learning techniques make use of CSD to detect the shift and then adapt to non-stationarities in the streaming EEG.

The parameter CR_{Thres} is used to decide whether the information in hand is useful or not. If the information is useful then it is added to the existing KB. The discarded information may come from a different distribution or it may have not provided much confidence to add into KB. The value of CR_{Thres} and K are important, and are required to be carefully selected in order to achieve superior performance. For instance, for the method $Trans_1$, the value of CR_{Thres} is empirically selected in the range [0.50-1]. In Trans2, the parameters are selected based upon the grid search method and the accuracy is superior for both of the datasets. This implies that the performance of the proposed method mostly depends upon the optimal choice of CR_{Thres}.

The experimental results demonstrated the effectiveness of the proposed covariate shift-detection and adaptation learning strategy. The results also showed that the proposed method with CSP filters and optimized parameters is significantly better than the traditional learning methods, and SSL with CSP filters. The combination of EWMA based covariate shift-detection and adaptive learning is thus a good choice for learning in non-stationary environments. The robustness of the CSD test plays an important role in initiating a correct adaptive action.

VI. CONCLUSION

The proposed methodology is a flexible tool for adaptive learning in non-stationary environments and effectively accounts for the effect of the covariate shifts. In this paper, two methods (TLCSD and ALCSD) are proposed for the covariate shift-adaptation using a two-stage covariate shift detection test. The CSD test in the first stage uses the SD-EWMA test; and in the second stage, the multivariate Hotelling's T-Square statistical hypothesis test is used. The CSD test is found very effective in detecting the covariate shifts in the data in real-time. Based on the detected significant shifts, the algorithm initiates adaptive corrective action. The performance of the proposed methods was

evaluated on multivariate cognitive task detection problem in the EEG-based BCIs simulated with BCI Competition IV datasets 2A & 2B, and superior results in terms of increased classification accuracy are obtained. The TLCSD and ALCSD have shown statistically significant improvement. This work is planned to be extended further by employing the CSD into the task of fault monitoring as well.

ACKNOWLEDGEMENT

H.R. was supported by Ulster University Vice-Chancellor's research scholarship (VCRS). G.P. and H.C. were supported by the Northern Ireland Functional Brain Mapping Facility project (1303/101154803), funded by InvestNI and the Ulster University. G.P. and H.R. were also supported by the UKIERI DST Thematic Partnership project "A BCI operated hand exoskeleton based neuro-rehabilitation system" (UKIERI-DST-2013-14/126).

References

Alippi, C., Boracchi, G. & Roveri, M., 2013. Just-In-Time Classifiers for Recurrent Concepts. *IEEE Transactions on Neural Networks and Learning Systems*, 24(4), pp.620–634.

Ang, K.K. et al., 2008. Filter Bank Common Spatial Pattern (FBCSP). In *Proc. Int'l Joint Conf. Neural Networks (IJCNN)*. pp. 2390–2397.

Ang, K.K. et al., 2012. Filter Bank Common Spatial Pattern Algorithm on BCI Competition IV Datasets 2a and 2b. *Frontiers in neuroscience*, 6, p.39. Available at: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3314883&tool=pmcentrez&rendertype=abstract [Accessed October 5, 2014].

Arvaneh, M., Cuntai, G., et al., 2013. Optimizing Spatial Filters by Minimizing Within-Class Dissimilarities in Electroencephalogram-Based Brain-Computer Interface. *IEEE Transactions on Neural Networks and Learning Systems*, 24(4), pp.610–619. Available at: http://europepmc.org/abstract/med/24808381 [Accessed June 24, 2014].

Arvaneh, M., Guan, C. & Chai Quek, 2013. EEG Data Space Adaptation to Reduce Intersession Nonstationary in Brain-Computer Interface. *Journal of Neural Computation*, 25, pp.1–26.

Bishop, C.M., 2006. *Pattern Recognition and Machine Learning*, New York: Springer.

Blankertz, B., Berlin BCI Competition. Available at: http://www.bbci.de/competition/.

Blankertz, B. et al., 2002. Classifying Single Trial EEG : Towards Brain Computer Interfacing. In *Advances in Neural Information Processing Systems*. pp. 157–164.

Blankertz, B. & Tomioka, R., 2008. Optimizing spatial filters for robust EEG single-trial analysis. *Signal Processing Magazine, IEEE*, pp.41–56. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4408441 [Accessed January 15, 2015].

Brunner, C., Leeb, R. & Müller-Putz, G., 2008. BCI Competition 2008–Graz data set A. , pp.1–6. Available at: http://www.bbci.de/competition/iv/desc_2a.pdf [Accessed October 6, 2014].

Buttfield, A., Ferrez, P.W. & Millán, J.D.R., 2006. Towards A Robust BCI: Error Potentials and Online Learning. *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, 14(2), pp.164–8. Available at: http://www.ncbi.nlm.nih.gov/pubmed/16792284.

Coyle, D., Prasad, G. & McGinnity, T.M., 2009. Faster Self-Organizing Fuzzy Neural Network Training and A Hyperparameter Analysis for A Brain-Computer Interface. *IEEE transactions on Systems, Man, and Cybernetics.*, 39(6), pp.1458–71.

Duda, R.O., Hart, P.E. & Stork., D.G., 2001. *Pattern Recognition*, Wiley-Interscience.

Elwell, R. & Polikar, R., 2011. Incremental Learning of Concept Drift in Non-Stationary Environments. *IEEE Transactions on Neural Networks*, 22(10), pp.1517–31.

Gama, J. et al., 2013. A Survey on Concept Drift Adaptation. *ACM Survey*, 1(1). Available at: http://www.win.tue.nl/~mpechen/publications/pubs/Gama_ACMCS_AdaptationCD_accepted.pdf [Accessed March 10, 2014].

Grossberg, S., 1988. Nonlinear Neural Networks: Principles, Mechanisms, and Architectures. *Neural Networks*, 1(1), pp.17–61. Available at: http://linkinghub.elsevier.com/retrieve/pii/0893608088900214.

Hotelling, H., 1947. Multivariate Quality Control-Illsutrated by the Air Testing of Sample Bombsights. *Techniques of Statistical Analysis*, pp.111–184.

Kelly, M.G., Hand, D.J. & Adams, N.M., 1998. The Impact of Changing Populations on Classifier Performance. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, pp. 367–371.

Kuncheva, L. & Faithfull, W., 2014. PCA feature extraction for change detection in multidimensional unlabeled data. *IEEE Transactions on Neural Networks and Learning Systems*, 25(1), pp.69–80. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6479367 [Accessed November 29, 2014].

Leeb, R. et al., 2008. BCI Competition 2008–Graz data set B. Graz University of Technology, Austria.

Leeb, R. et al., 2007. Brain-computer communication: motivation, aim, and impact of exploring a virtual apartment. *IEEE transactions on neural systems and rehabilitation engineering : a publication of the IEEE Engineering in Medicine and Biology Society*, 15(4), pp.473–82. Available at: http://www.ncbi.nlm.nih.gov/pubmed/18198704.

Li, Y. et al., 2010. Application of Covariate Shift Adaptation Techniques in Brain-Computer Interfaces. *IEEE Transaction on Biomedical Engineering*, 57(6), pp.1318–24.

Lotte, F. et al., 2007. A Review of Classification Algorithms for EEG-Based Brain-Computer Interfaces. *Journal of Neural Engineering*, 4(2), pp.1–13.

Mitchell, T., 1997. *Machine Learning*, McGraw Hill.

Müller, K., Krauledat, M. & Dornhege, G., 2004. Machine learning techniques for brain-computer interfaces. *Journal of Biomedical Engineering*, 49, pp.11–22. Available at: http://eprints.pascal-network.org/archive/00000416/ [Accessed January 15, 2015].

Ramoser, H., Müller-Gerking, J. & Pfurtscheller, G., 2000. Optimal spatial filtering of single trial EEG during imagined hand movement. *IEEE transactions on Rehabilitation Engineering*, 8(4), pp.441–446.

Raza, H., Prasad, G. & Li, Y., 2014. Adaptive learning with covariate shift-detection for non-stationary environments. In *IEEE 14th UK Workshop on Computational Intelligence (UKCI)*. Bradford, UK, pp. 1–8. Available at: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6930161.

Raza, H., Prasad, G. & Li, Y., 2013a. Dataset Shift Detection in Non-stationary Environments Using EWMA Charts. In *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. pp. 3151–3156. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6722290 [Accessed January 5, 2015].

Raza, H., Prasad, G. & Li, Y., 2013b. EWMA Based Two-Stage Dataset Shift-Detection in Non-stationary Environments. In *Artificial Intelligence Applications and Innovations (AIAI).* Springer Berlin Heidelberg, pp. 625–635.

Raza, H., Prasad, G. & Li, Y., 2015. EWMA model based shift-detection methods for detecting covariate shifts in non-stationary environments. *Pattern Recognition*, 48(3), pp.659–669.

Rezaei, S. et al., 2006. Different classification techniques considering brain computer interface applications. *Journal of neural engineering*, 3(2), pp.139–144. Available at: http://www.ncbi.nlm.nih.gov/pubmed/16705270 [Accessed October 28, 2014].

Rosenstiel, W., Bogdan, M. & Sp, M., 2012. Principal Component Based Covariate Shift Adaption to Reduce Non-Stationarity in a MEG-Based Brain-Computer Interface. *EURASIP Journal on Advances in Signal Processing*, pp.2–8.

Satti, A. et al., 2010. A Covariate Shift Minimisation Method to Alleviate Non-stationarity Effects for an Adaptive Brain-Computer Interface. In *20th International Conference on Pattern Recognition (ICPR)*. Ieee, pp. 105–108.

Shahid, S. & Prasad, G., 2011. Bispectrum-based feature extraction technique for devising a practical brain-computer interface. *Journal of neural engineering*, 8(2), p.025014. Available at: http://www.ncbi.nlm.nih.gov/pubmed/21436530 [Accessed November 13, 2014].

Shimodaira, H., 2000. Improving Predictive Inference Under Covariate Shift by Weighting the Log-Likelihood Function. *Journal of Statistical Planning and Inference*, 90(2), pp.227–244.

Sugiyama, M., 2007. Covariate Shift Adaptation by Importance Weighted Cross Validation. *Journal of Machine Learning Research*, 8, pp.985–1005.

Sugiyama, M., 2012. Learning under Non-stationarity : Covariate Shift Adaptation by Importance Weighting. In *Handbook of Computational Statistics: Concept and Methods*. Springer, pp. 1–27.

Suk, H. & Lee, S., 2013. A novel Bayesian framework for discriminative feature extraction in brain-computer interfaces. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 35(2), pp.286–299. Available at: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6175024 [Accessed October 31, 2014].

Tangermann, M. et al., 2012. Review of the BCI Competition IV. *Frontiers in neuroscience*, 6, p.55. Available at: http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3396284&tool=pmcentrez&rendertype=abstract [Accessed March 21, 2014].

Thulasidas, M. et al., 2006. Robust Classification of EEG Signal for Brain Computer Interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(1), pp.24–29.

Vapnik, V., 1999. An Overview of Statistical Learning Theory. *IEEE Transactions on Neural Networks*, 10(5), pp.988–99.

Vidaurre, C. et al., 2006. A Fully On-line Adaptive BCI. *IEEE Transaction on Biomedical Engineering*, 53(6), pp.1214–1219.

Wolpaw, J.R. et al., 2002. Brain-Computer Interfaces for Communication and Control. *Clinical Neurophysiology : Official Journal of the International Federation of Clinical Neurophysiology*, 113(6), pp.767–91.

Zhu, X., 2008. *Semi-supervised learning literature survey*, Computer Science Technical Report 1530, University of Wisconsin-Madison. Available at: http://www.loni.ucla.edu/~ztu/courses/2013_CS_spring/reading/ssl_survey.pdf [Accessed January 20, 2014].

Zhu, X. & Ghahramani, Z., 2002. *Learning from Labeled and Unlabeled Data with Label Propagation*, Technical Report CMU-CALD-02-107, Carnegie Mellon University.